Amendments to the Claims:

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

1. - 20. (canceled)

21. (currently amended) A method for facilitating the reuse of computer graphics models in a computer system comprising a processor and a memory, the method comprising:

receiving, from a first user in a plurality of users, one or more commands for creating a first computer graphics model in a model creation environment in the processor, wherein the first computer graphics model includes a first plurality of geometric objects;

receiving, from the first user, a request to ~~include an~~ create an instance of a second computer graphics model [[in]] within the first computer graphics model in the processor, wherein the second computer graphics model includes a second plurality of geometric objects, and wherein the second computer graphics model is independent of the first computer graphics model;

in response to the request, retrieving a specification of the second computer graphics model from the memory, wherein the specification of the second computer graphics model ~~including~~ includes information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are overridable;

creating the instance of the second computer graphics model ~~and including said instance in~~ within the first computer graphics model in the processor; and

for each object in the instance of the second computer graphics model:

determining, based on the specification of the second computer graphics model, attributes of ~~said each object~~ specific geometric objects from the second plurality of geometric objects that are overridable in the processor; and

enabling the first user to specify an override ~~values for the attributes of said each object~~ value for an attribute of a specific geometric object that [[are]] is determined to be overridable in the processor; and

storing a reference to the second computer graphics model in the first computer graphics, wherein the second computer graphics model is not stored in the first computer graphics model.

22. (previously presented) The method of claim 21, wherein the first plurality of geometric objects in the first computer graphics model are organized according to a hierarchical structure, and wherein including the instance of the second computer graphics model in the first computer graphics model comprises geometrically coupling the instance of the second computer graphics model to the hierarchical structure.

23. (currently amended) The method of claim 21 further comprising:

receiving, from the first user, [[a new]] the override value for an overridable attribute of an object in the instance of the second computer graphics model in the processor;

applying the [[new]] override value as a current value for the overridable attribute in the context of the first computer graphics model in the processor ; and

storing a specification of the first computer graphics model wherein the specification includes [[a]] the reference to the specification of the second computer graphics model and the [[new]] override value in the memory.

24. (previously presented) The method of claim 23, wherein the specification of the first computer graphics model is stored as a first file, and wherein the specification of the second computer graphics model is stored as a second file distinct from the first file.

25. (currently amended) The method of claim 23 further comprising:

receiving, from a second user in the plurality of users, one or more commands for creating a third computer graphics model in the model creation environment in the processor, the third computer graphics model including a third plurality of objects;

receiving, from the second user, a request to include an instance of the first computer graphics model in the third computer graphics model in the processor;

in response to the request, retrieving the specification of the first computer graphics model from the memory, the specification of the first computer graphics model including information identifying, for at least one object in the first plurality of geometric objects, one or more attributes of said at least one object that are overridable;

creating the instance of the first computer graphics model ~~and including said instance in~~ within the third computer graphics model in the processor; and

for each object in the instance of the first computer graphics model:

determining, based on the specification of the first computer graphics model, attributes of said each object that are overridable in the processor; and

enabling the second user to override values for the attributes of said each object that are determined to be overridable in the processor

storing a reference to the first computer graphics model in the third computer graphics, wherein the first computer graphics model is not stored in the third computer graphics model.

26. (previously presented) The method of claim 21, wherein the second computer graphics model is created by a second user in the plurality of users distinct from the first user.

27. (currently amended) The method of claim 26 further comprising:

receiving, from the first user, a request to include an instance of a third computer graphics model in the first computer graphics model in the processor, the third computer graphics model having been created by a third user in the plurality of users distinct from the first and second users, the third computer graphics model including a third plurality of objects;

in response to the request, retrieving a specification of the third computer graphics model from the memory, the specification of the third computer graphics model including information identifying, for at least one object in the third plurality of objects, one or more attributes of said at least one object that are overridable;

creating the instance of the third computer graphics model ~~and including said instance in~~ within the first computer graphics model in the processor; and

for each object in the instance of the third computer graphics model:

determining, based on the specification of the third computer graphics model, attributes of said each object that are overridable in the processor; and

enabling the first user to override values for the attributes of said each object that are determined to be overridable in the processor; and

storing a reference to the third computer graphics model in the first computer graphics, wherein the third computer graphics model is not stored in the first computer graphics model.

28. (currently amended) The method of claim 21, wherein the specification of the second computer graphics model further includes information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are not overridable, and

wherein the method further comprises:

for each object in the instance of the second computer graphics model:

determining, based on the specification of the second computer graphics model, attributes of said each object that are not overridable in the processor; and

preventing the first user from overriding values for the attributes of said each object that are determined to be not overridable in the processor.

29. (currently amended) A system for facilitating the reuse of computer graphics models, the system comprising:

a storage device configured to store specifications for a plurality of computer graphics models; and

a processing component in communication with the storage device, the processing component being configured to:

receive, from a first user in a plurality of users, one or more commands for creating a first computer graphics model in a model creation environment, wherein the first computer graphics model includes a first plurality of geometric objects;

receive, from the first user, a request to include an instance of a second computer graphics model in the first computer graphics model, wherein the second computer graphics model includes a second plurality of geometric objects, and wherein the second computer graphics model is independent of the first computer graphics model;

in response to the request, retrieve a specification of the second computer graphics model from the storage device, the specification of the second computer graphics model including information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are overridable;

create the instance of the second computer graphics model and include said instance in the first computer graphics model; and

for each object in the instance of the second computer graphics model:

determine, based on the specification of the second computer graphics model, attributes of said each object that are overridable; and

enable the first user to override values for the attributes of said each object that are determined to be overridable ;

wherein a reference to the second computer graphics model from the storage device in stored in the first computer graphics model, and

wherein the second computer graphics model is not stored in the first computer graphics model.

30. (previously presented) The system of claim 29, wherein the first plurality of geometric objects in the first computer graphics model are organized according to a hierarchical structure, and wherein including the instance of the second computer graphics model in the first computer graphics model comprises geometrically coupling the instance of the second computer graphics model to the hierarchical structure.

31. (previously presented) The system of claim 29, wherein the processing component is further configured to:

receive, from the first user, a new value for an overridable attribute of an object in the instance of the second computer graphics model;

apply the new value as a current value for the overridable attribute in the context of the first computer graphics model; and

store a specification of the first computer graphics model in the storage device, wherein the specification includes a reference to the specification of the second computer graphics model and the new value.

32. (previously presented) The system of claim 31, wherein the specification of the first computer graphics model is stored as a first file in the storage device, and wherein the specification of the second computer graphics model is stored as a second file in the storage device distinct from the first file.

33. (previously presented) The system of claim 31, wherein the processing component is further configured to:

receive, from a second user in the plurality of users, one or more commands for creating a third computer graphics model in the model creation environment, the third computer graphics model including a third plurality of objects;

receive, from the second user, a request to include an instance of the first computer graphics model in the third computer graphics model;

in response to the request, retrieve the specification of first computer graphics model from the storage device, the specification of the first computer graphics model including information identifying, for at least one object in the first plurality of geometric objects, one or more attributes of said at least one object that are overridable;

create the instance of the first computer graphics model and include said instance in the third computer graphics model; and

for each object in the instance of the first computer graphics model:

determine, based on the specification of the first computer graphics model, attributes of said each object that are overridable; and

enable the second user to override values for the attributes of said each object that are determined to be overridable.

34. (previously presented) The system of claim 29, wherein the second computer graphics model is created by a second user in the plurality of users distinct from the first user.

35. (previously presented) The system of claim 34, wherein the processing component is further configured to:

receive, from the first user, a request to include an instance of a third computer graphics model in the first computer graphics model, the third computer graphics model having been created by a third user in the plurality of users distinct from the first and second users, the third computer graphics model including a third plurality of objects;

in response to the request, retrieve a specification of the third computer graphics model from the storage device, the specification of the third computer graphics model including information identifying, for at least one object in the third plurality of objects, one or more attributes of said at least one object that are overridable;

create the instance of the third computer graphics model and include said instance in the first computer graphics model; and

for each object in the instance of the third computer graphics model:

determine, based on the specification of the third computer graphics model, attributes of said each object that are overridable; and

enable the first user to override values for the attributes of said each object that are determined to be overridable.

36. (previously presented) The system of claim 29, wherein the specification of the second computer graphics model further includes information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are not overridable, and

wherein the processing component is further configured to:

for each object in the instance of the second computer graphics model:

determine, based on the specification of the second computer graphics model, attributes of said each object that are not overridable; and

prevent the first user from overriding values for the attributes of said each object that are determined to be not overridable.

37. (previously presented) A tangible machine-readable medium for a computer system, the tangible machine readable medium having stored thereon a series of instructions which, when executed by a processing component, cause the processing component to:

receive, from a first user in a plurality of users, one or more commands for creating a first computer graphics model in a model creation environment, wherein the first computer graphics model includes a first plurality of objects;

receive, from the first user, a request to include an instance of a second computer graphics model in the first computer graphics model, wherein the second computer graphics model includes a second plurality of objects, and wherein the second computer graphics model is independent of the first computer graphics model;

in response to the request, retrieve a specification of the second computer graphics model, the specification of the second computer graphics model including information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are overridable;

create the instance of the second computer graphics model and including said instance in the first computer graphics model; and

for each object in the instance of the second computer graphics model:

determine, based on the specification of the second computer graphics model, attributes of said each object that are overridable; and

enable the first user to override values for the attributes of said each object that are determined to be overridable.

38. (previously presented) The tangible machine-readable medium of claim 37, wherein the first plurality of geometric objects in the first computer graphics model are organized according to a hierarchical structure, and wherein including the instance of the second computer graphics model in the first computer graphics model comprises geometrically coupling the instance of the second computer graphics model to the hierarchical structure.

39. (previously presented) The tangible machine-readable medium of claim 37, wherein the series of instructions further cause the processing component to:

receive, from the first user, a new value for an overridable attribute of an object in the instance of the second computer graphics model;

apply the new value as a current value for the overridable attribute in the context of the first computer graphics model; and

store a specification of the first computer graphics model, wherein the specification includes a reference to the specification of the second computer graphics model and the new value.

40. (previously presented) The tangible machine-readable medium of claim 39, wherein the specification of the first computer graphics model is stored as a first file, and wherein the specification of the second computer graphics model is stored as a second file distinct from the first file.

41. (previously presented) The tangible machine-readable medium of claim 39, wherein the set of instructions further cause the processing component to:

receive, from a second user in the plurality of users, one or more commands for creating a third computer graphics model in the model creation environment, the third computer graphics model including a third plurality of objects;

receive, from the second user, a request to include an instance of the first computer graphics model in the third computer graphics model;

in response to the request, retrieve the specification of the first computer graphics model, the specification of the first computer graphics model including information identifying, for at least one object in the first plurality of geometric objects, one or more attributes of said at least one object that are overridable;

create the instance of the first computer graphics model and include said instance in the third computer graphics model; and

for each object in the instance of the first computer graphics model:

determine, based on the specification of the first computer graphics model, attributes of said each object that are overridable; and

enable the second user to override values for the attributes of said each object that are determined to be overridable.

42. (previously presented) The tangible machine-readable medium of claim 37, wherein the second computer graphics model is created by a second user in the plurality of users distinct from the first user.

43. (previously presented) The tangible machine-readable medium of claim 42, wherein the set of instructions further cause the processing component to:

receive, from the first user, a request to include an instance of a third computer graphics model in the first computer graphics model, the third computer graphics model having been created by a third user in the plurality of users distinct from the first and second users, the third computer graphics model including a third plurality of objects;

in response to the request, retrieve a specification of the third computer graphics model, the specification of the third computer graphics model including information identifying, for at least one object in the third plurality of objects, one or more attributes of said at least one object that are overridable;

create the instance of the third computer graphics model and include said instance in the first computer graphics model; and

for each object in the instance of the third computer graphics model:

determine, based on the specification of the third computer graphics model, attributes of said each object that are overridable; and

enable the first user to override values for the attributes of said each object that are determined to be overridable.

44. (previously presented) The tangible machine-readable medium of claim 37, wherein the specification of the second computer graphics model further includes information identifying, for at least one object in the second plurality of geometric objects, one or more attributes of said at least one object that are not overridable, and

wherein the method further comprises:

for each object in the instance of the second computer graphics model:

determining, based on the specification of the second computer graphics model, attributes of said each object that are not overridable; and

preventing the first user from overriding values for the attributes of said each object that are determined to be not overridable.